FERMILAB-PUB-04-268-E

The BTeV Trigger and Data Acquisition System

J. N. Butler

Fermilab, Box 500, Batavia Illinois, US
butler@fnal.gov

Abstract

The BTeV trigger inspects every beam crossing of the Fermilab Tevatron, running at a luminosity of $2x10^{32}/cm^2$-s, and selects events that have "detached vertices" from B decays occurring downstream of the main interaction. The system uses a massively parallel system of FPGAs and microprocessors to produce a trigger decision on average every 396 ns. The trigger calculations are facilitated by the 23 Million channel pixel detector that provides the input to the trigger. Front end electronics sparsifies the remainder of event data and sends it to large, Tbyte, memory buffers that store it until the trigger decision can be made. This complex system presents special challenges in fault monitoring and power and cooling.

I. Introduction and Overview of BTeV

BTeV [1] is an experiment that will be carried out at the Fermilab Tevatron Collider starting in 2009. It will run in the C0 interaction region. Its goal is to study CP Violation and Mixing in the decays of particles containing bottom and charm quarks.

The Standard Model of Particle Physics fails to explain the amount of matter in the Universe. If matter and antimatter did not behave slightly differently, all the matter and antimatter in the early universe would have annihilated into pure energy and there would be no baryonic matter (protons, neutrons, nuclei). The Standard Model of Particle Physics accommodates matter-antimatter asymmetry in K meson and B meson decays, but it predicts a universe that contains about 1/10,000 of the density of baryonic matter we actually observe. Looking among the B decays for new sources of matter-antimatter asymmetry that resolve this serious deficiency is the goal of the next round of B experiments, including BTeV.

The Tevatron, running at a luminosity of $2x10^{32}/cm^2$-s, produces $4x10^{11}$ b-hadrons per year of operation. These include all species of b-hadrons, including B_d, B_u, and B_s mesons as well as b-baryons of all kinds. This permits studies that extend the work at e^+e^- colliders that only produce B_d and B_u. However, to exploit the large number of B's a dedicated experiment optimised for CP studies needs to be constructed.

II. Expected Running Conditions

Table 1 shows the operating conditions expected during BTeV. The physics goals of the experiment, taken together with these conditions, define the task for the BTeV trigger system. The number of interactions per second results in a data volume that is too large to be stored on archival data for analysis. The trigger [2][3] must select from this huge rate the approximately 1000 events per second that contain b-hadrons and enter the spectrometer. This is a significant challenge.

Table 1: Operating Properties of the Tevatron during BTeV

Luminosity	$2x10^{32}$
# interactions/s	$15x10^6$
# of B-anti B pairs/10^7 s	$2x10^{11}$
#of B events per background event	1/500 (only 1/500,000 are "interesting" B decays)
Bunch spacing	396 ns (originally 132 ns)
Luminous region length	σ_z = 30 cm
Luminous region radius	$\sigma_x \sim \sigma_y \sim 30$ μm
#Interactions/beam crossing	<6.0>

III. The Physics Basis of the Trigger

To form a trigger, we must exploit properties of events with B-hadrons that differentiate them from the much larger number of ordinary or "minimum bias" events. Figure 1 illustrates the key characteristic that distinguishes B-events. The B's produced in the interaction travel a short distance, between a few tenths of a mm and a few mm from the point of the interaction and then decay into two or more (typically 5) particles. The presence of these "detached vertices" or "secondary vertices" is the signature of a B event and the best way to trigger on them. However, this requires the trigger to reconstruct tracks and assemble the tracks into vertices to find the events with evidence of detached vertices. This task must be done in quasi-real time so that a decision must be made on average every 396 ns. This represents a formidable challenge that has not been achieved yet in particle physics.

Conventional high energy physics triggers are usually based on a three level hierarchy. The lowest level, which we will refer to as Level 1, uses fairly simple signals to form triggers

usually within a fixed amount of time, typically a few microseconds. They chose events within this limited time budget based on relatively simple criteria, such as various sums of calorimeter pulse heights. This reduces the data rate so that Level 2 has more time to spend on each remaining event. Level 2 usually is a mixture of dedicated trigger hardware and computing elements. The Level 2 trigger further reduces the rate providing a relatively small sample of events to the Level 3 trigger that now has enough time to process the events in a massively parallel farm of microprocessors using algorithms that are quite similar to a full offline analysis to make the final decision to discard the event or write it to archival storage for offline physics studies.

BTeV also has a three level trigger hierarchy [4]. The main difference is that massive computing is applied at Level 1. The challenge for the BTeV trigger and data acquisition system is to reconstruct particle tracks and interaction vertices for EVERY interaction that occurs in the BTeV detector, and to select interactions with *B* decays.

The trigger performs this task using 3 stages, referred to as Levels 1, 2 and 3:

"L1" – looks at every interaction, and rejects at least 98% of background based on full track and vertex reconstruction using a silicon pixel detector described below;

L2" – uses L1 results and performs more refined analyses for data selection; and

"L3" – performs a complete analysis using all of the data for an interaction.The total effect of the trigger is to reject > 99.8% of background. Keep > 50% of *B* events.

The Data Acquisition System (DAQ) [5] saves all of the detector data in memory for as long as is necessary for Level 1 to analyse each interaction (~ 0.5 millisecond on average for L1), and moves data to L2/3 processing units and archival storage for selected interactions.

The key ingredients that make it possible to meet this challenge:

- BTeV pixel detector [6] with its exceptional pattern recognition capabilities; and
- Rapid development in technology and lower costs for – FPGAs, microprocessor CPUs, and memory.

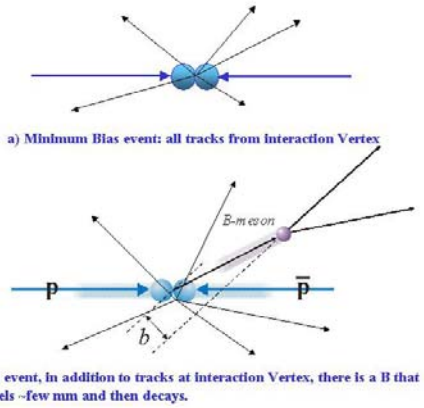

Figure 1: A schematic representation of A) "ordinary events", containing only the interaction vertex, and B) events containing a B-hadron showing a detached vertex from a B-meson that eventually decays into two particles.

IV. THE BTEV SPECTROMETER AND PIXEL DETECTOR

To achieve such an ambitious goal, one needs to design the spectrometer specifically with this triggering problem in mind. A schematic of the BTeV spectrometer is shown in Fig. 2.

The most important features related to the trigger are:A precision vertex detector of planar pixel arrays located right near the IR. This provides sufficient track resolution to separate the various vertices. The pixel detector position resolution is of order 6 microns.

- The pixel detector is located in the middle of a large dipole magnet, also centred on the IR. It produces measurements that enable the trigger to determine the momentum of charged tracks that traverse the detector. This is essential because it allows the trigger to eliminate from its calculations very low momentum tracks that can be badly scattered and appear to be detached from the primary vertex. These tend to result in "fake" triggers. The decay products of B events are generally high momentum particles.
- A vertex trigger at the lowest level of the trigger system that can select events based on evidence for detached vertices.
- A very high speed, high capacity data acquisition system that is capable of recording every B event that is selected by the trigger without exercising further judgment as to the exact topology or "physics value" of the B decay. This gives BTeV the widest possible range of B physics to analyse and avoids any bias towards designing an experiment optimised for today's fashionable decays but that might not be efficient on many

types of decays that will be interesting when the experiment actually takes data.

In order to carry out tracking and vertex calculations at very high rates with an affordable amount of hardware, one needs to provide the trigger system with the best possible tracking information in a form that eases the task of pattern recognition. BTeV has chosen to develop a high speed, high rate precision tracker based on silicon pixel detectors. The detector, shown schematically in Fig. 3, has 30 stations of pixels distributed along the IR. The pixels are rectangles of 50 μm X 400 μm. Each station consists of two views, one measuring X with high precision and Y with lower precision and the second measuring Y with high precision and X with lower precision. This technology is chosen because it gives essentially 3-dimensional space points; It has excellent spatial resolution of 5-10 microns depending on the angle of the track as it traverses the plane of the pixel detector; a very low occupancy of 10^{-4}; a very fast signal that ends well before the next beam crossing; and radiation hardness that permits it to survive very close to the beam, a necessary condition for excellent vertex resolution. While pixel detectors of comparable complexity are being developed for other detectors, including CMS and ATLAS at the LHC, the BTeV pixel detector is unique in that it is used directly in the lowest level of the trigger and that each of the 23 million pixels has its own 3 bit flash ADC. This allows us to exploit charge sharing to improve the spatial resolution. Excellent spatial resolution helps the pixel detector measure the curvature of tracks so that the momentum can be calculated at the trigger level. The whole system is digitised, sparsified, and read out into the trigger system at the beam-crossing rate.

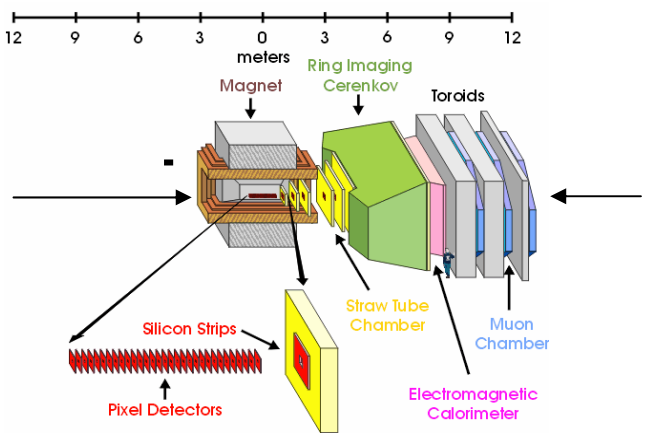

Figure 2: Schematic of the BTeV Spectrometer. The magnet is centred on the Interaction Region (IR). The pixel detector is also centred on the IR. Because it is immersed in a magnetic field it can reconstruct particle momenta for use in the trigger.

The near-3D space points returned by the pixel detector make pattern recognition very simple and reduce the amount of computing time needed to carry out tracking and vertex calculations two orders of magnitude relative to a silicon strip detector. The high quality inputs make the trigger calculations possible with a reasonable number of processors.

V. The BTeV Front End Electronics and Data Acquisition System

The trigger system actually deals with "beam crossings", treating each crossing as a separate computing problem and trying to determine whether any of the interactions are B events. Since the crossings have a variable number of interactions and the individual interactions have varying complexity, the time it takes to compute for an individual crossing is highly variable. In order to keep all processing elements busy, BTeV's trigger and DAQ have

- no fixed latency at any level. Decisions are made in variable amounts of time and transmitted as soon as they are known; and
- no requirement of time ordering. It is common for system to be carrying out computations on a crossing while it has already completed several later ones.

This is turn requires massive amounts of buffering throughout the system. To limit the amount of data that needs to be buffered, on-the-fly sparsification (zero suppression) in the front ends is implemented. By sparsifying the data and shipping it out every 396 ns, the front ends keep the data volume from the very large number of channels from the pixel detector and all the other BTeV detectors manageable. The DAQ must store the sparsified data from all the detectors for as long as it takes to make the Level 1 trigger decision. This is done by sending the sparsified data to a large buffer memory system that is based on commercial PC memory whose cost is low and continues to decrease. BTeV plans to have about 1 Tbyte of buffer memory, an amount big enough to hold nearly 1 second of beam crossing data. This amount of time is much longer than the average Level 1 trigger time of 0.5 milliseconds. Once the Level 1 trigger makes a decision, the 98-99% of the crossings that fail the trigger are erased from the buffers, freeing the memory for other events. The 1-2% that pass are moved to other buffers for Level 2/3 processing. Since the amount of data is vastly reduced, it is possible to store crossings that have passed Level 1 for very long amounts of time while the Level 2/3 calculations are being performed.

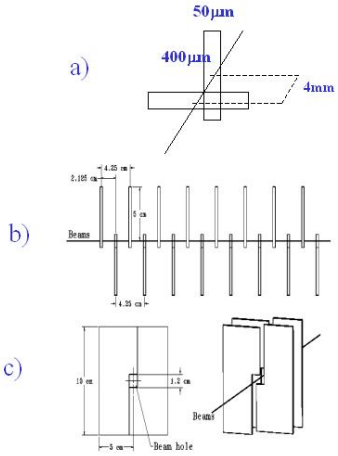

Figure 3: Schematic of BTeV Pixel Detector: A) shows the crossed rectangular pixels in a single station. Each station provides a high precision and lower (but still very good) precision measurement of both X and Y; B) shows the layout along the beam (in Z); and 3) shows the layout transverse to the beam. The detector is only 10cm x 10cm in cross section, occupies about 1.2 m along the beam, and has 23 Million pixels. It has a 12mm X 12mm hole in the center that the beams pass through. The whole system is under vacuum.

The electronics to accomplish the sparsification is a combination of ASICs designed specifically for BTeV and ASICs that have been used in other experiments, in some cases with modifications. All these detectors operate fairly close to the colliding beams and are required to have various degrees of radiation tolerance.

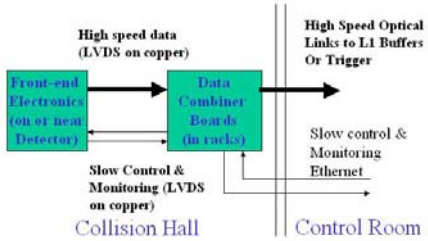

Figure 4: Schematic of Front End electronics and connection to the BTeV Control Room

The pixel detector is read out by a custom chip, FPIX2, designed at Fermilab that is bump bonded to the pixel sensors. It is realized in 0.25 μm CMOS technology that has been demonstrated to be radiation hard. The other chips and channel counts for all BTeV detectors are shown in Table 2.

Table 2: Front end electronics chips, chip technologies, and counts for BTeV detectors

Detector	# of Channels	Chip Name	Process	Channels Per chip	# of chips
Pixel	23 million	FPIX	0.25 μm CMOS	22x128 pixels	8100
Silicon Strip	117,000	FSSR	0.25 μm CMOS	22 x128	1008
Straw Detector	53,528	ASDQ	MAXIM SHPi Analog Bipolar	8	6696
Straw Detector	53,528	TDC	0.25 μm CMOS	24	2232
Muon Detector	36,864	ASDQ	MAXIM SHPi	8	2232
RICH	144,256+ 5,048	RICH hybrid		128	1196+ 80
RICH		RICH MUX		128	332+ 24
EMCAL	10,100	QIE9	AMS 0.8 μm BICMOS	1	10,100

Data is transported from the front ends by a BTeV-specific device, the Data Combiner Board (DCB) that serializes the data and sends it to the Control Room over fibre optic links. The data are stored in the Level 1 Buffer System (L1B) while the trigger is making its decision. The data path from the front end to the control room is shown in Figure 4.

VI. First Level Trigger Implementation

The first level trigger is based on the pixel detector. Figure 5 is a schematic of the first portion of the electronics. The pixel processor collects hits from the same crossing (time ordering), applies a clustering algorithm, and produces a coordinate for each cluster. It passes the list of coordinates to the segment finder that executes the first part of the tracking algorithm.

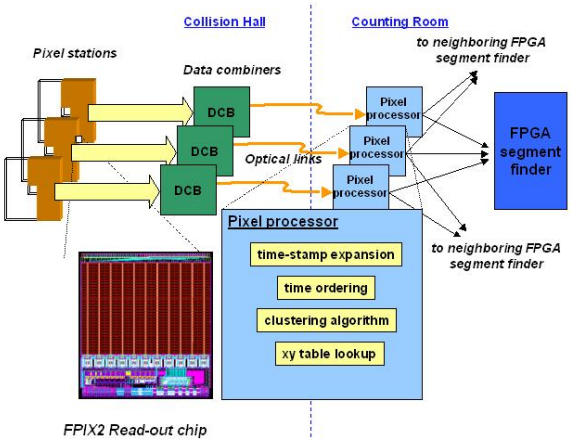

Figure 5: Pixel trigger electronics, showing the pixel detector half-stations, the pixel readout chip, the DCBs, the optical fibres to the counting room and the pixel processor.

The trigger algorithm has two major stages:

- Segment finding; and
- Track/vertex finding.

A. Segment finding

Pixel hits from three neighboring stations are used to find the beginning and ending segments of tracks. These three station segments are called triplets. An "inner triplet" is associated with a track as it enters the pixel detector from the interaction region and represents the start of the track. Since nearly all tracks entering the pixel detector this way and that will enter the forward spectrometer have a hit in the first centimetre of the pixel detector, only that limited region is used to "seed" or initiate searches for triplets. This greatly reduces the number of calculations that have to be performed. Similarly, an "outer triplet" is associated with a track as it leaves the pixel detector, either through the side or the front or rear faces. An "outer triplet" represents the end of the track in the pixel detector. Again, nearly all outer triplets start very close to the detector boundary so only a limited region is used to seed the search for outer triplets. Figure 6 shows the regions and hits used in the actual segment finding calculations.

Figure 6: A cross section of the pixel detector, showing the regions used for seeding "inner" and "outer" segments. Outer segments may also exit through the upstream or downstream faces so regions near those faces are also used to seed triplet searches.

The segment finding algorithm is very standard and works as follows:

1. First, starting with a seed hit in the "inner region" of plane N-1, one projects a cone onto plane N that corresponds to a range of legitimate and interesting tracks that would fall within the pixel detector acceptance;
2. For each hit, "I_N" within this range, one projects from this hit and the seed back to the Z position of plane N-2. If the projection falls within pixel plane N-2, then the seed is not the first point on an inner segment with hit I_N. One advances to the next hit in plane N;
3. If the projection falls inside the beam hole in the pixels at station N-2 instead, then one projects the seed and hit I_N into pixel plane N+1; if a confirming hit "J" is found, this seed, I_N, and J_{N+1} are an "inner segment."

Figure 6 shows the inner and outer regions and shows a typical event. The tracks and the inner and outer segments can be seen very clearly by eye.

The opportunities for parallelism are evident. Individual crossing must be handled separately. Each of the 30 stations can be a seed and each station can be handled in conjunction with the two adjacent downstream ones as a separate problem. Finally, as separate search can be done from each plane for tracks pointing downstream (towards the instrumented side of BTeV) or upstream (towards the un-instrumented side of BTeV). Both sets of tracks are useful in determining the primary vertices. Outer segment finding is done in the same way and in parallel. In the bend view, both inner and outer segments are found. These will eventually be matched and the difference in directions between an inner segment and its outer matching segment will give a measurement of the momentum. In the non-bend view, segment finding is done in parallel with the bend view, but only inner segments are searched for since they provide enough information to measure the track horizontal position and angle to extrapolate it back to the interaction vertex.

The segment finding algorithm is implemented with a system of 480 FPGAs. This number is based on a prototype implementation of the algorithm for an Altera [7] EPC20K1000 FPGA. Our work shows that the current design will fit comfortably in various devices offered by Altera and that similar devices from Xilinx [8] can be used with minor changes to the code.

Segment finding FPGAs do their tasks whenever hit data are available. Segments for several different crossings are being generated all at one.

B. Track and Vertex Finding

The next stage or processing involves delivering all the segments associated with a single beam crossing to one CPU in the track/vertex finding processor farm. A sorting switch sends all the segment data from one crossing to a single CPU. Buffering in both the switch and the CPU are required so that the processor knows that it has all the data before it begins to work on a crossing. The processor then does segment matching to form tracks and applies an algorithm to find "primary interaction vertices." Vertex finding constitutes projecting found tracks back into the interaction region and clustering them. Since tracks from B decays tend to have somewhat higher transverse momentum relative to the beam direction than tracks from the main interaction vertex, a requirement is placed on the tracks used in the clustering that they be below a certain transverse momentum. Typically several interaction vertices are found in each crossing, but they are usually quite well separated due to the length of the Tevatron luminous region. Each track not falling into these clusters and whose transverse momentum is above some value (typically 300 MeV/c) is extrapolated back to the nearest interaction vertex and its impact parameter, b, relative that vertex and its associated uncertainty, σ_b, are calculated. The quantity b/σ_b is used to evaluate detachment. A value of $b/\sigma_b>3$ is currently taken as the requirement to call a track "detached." The primary Level 1 trigger currently requires TWO tracks detached with respect to the same primary vertex to meet the criteria for a "Level 1 ACCEPT."

The processors that carry out the tracking/vertexing part of the Level 1 trigger were originally going to be Digital Signal Processors, due to budget considerations. However, the falling costs of conventional microprocessors now make it possible to employ them. Our system prototype is done with Apple G5 processors but there are INTEL-style processors that can also meet our requirements for the final system.

The hardware architecture of the Level 1 system is shown in Fig. 7.

The current Level 1 Tracking and Vertexing algorithm is written in the C Programming Language. It takes an average of 379μs/crossing on a 2.0 GHz Apple G5 [9] for minimum bias events with an average of 6 interactions per crossing. We assume that when BTeV runs, we will have the equivalent of a dual core 4 GHz G5 as a basic processor unit. Extrapolating the current performance, adding time for input buffer manipulations and fault monitoring, and providing a 40% extra capacity, we estimate that we need 528 dual-core 4 GHz G5 CPUs for the entire system, or about 264 dual-CPU processing nodes. We, of course, will continue to refine and optimise the code and will continue to test a wide variety of CPUs.

C. Hardware Implementation

Over the years of development, BTeV has replaced many custom or complex pieces of the trigger with commercial hardware. The custom "sorting switch" at Level 1 will be implemented instead with a commercial Infiniband switch [10]. The DSPs have now been replaced by conventional CPUs. A possible realization of the BTeV Level 1 Trigger is shown in Fig. 8.

D. Global Level 1

The actual Level 1 trigger decision is more complicated than indicated above. First, in addition to the pixel detector-based detached vertex trigger, BTeV has a stand-alone Level 1 Muon Trigger. This system uses much of the same hardware as the detached vertex trigger, but processes data coming only from the muon proportional tubes at the downstream end of BTeV. Many B decays, including some of the most important ones, involve two muons in the final state. This trigger somewhat enhances the efficiency for recording those, although most of them satisfy the detached vertex trigger. Most importantly, it provides a good independent monitor of the detached vertex trigger. Second, in order to help monitor the performance of the detached vertex trigger, we also take several types of triggers with various requirements relaxed to be able to see how the efficiency evolves as a function of our selection criteria. These triggers may have to be prescaled. Third, we want to collect a variety of calibration triggers some based on the special triggers for individual detectors and some even based on non-beam related signals such as LEDs driven by pulse generators.

The generation of the actual Level 1 trigger is handled by a subsystem called "global Level 1" (GL1) that makes the final decision on whether to declare that a beam crossing is in fact to be rejected or to be ACCEPTED and passed to higher trigger levels for additional processing and evaluation. Global Level 1 is a processor farm of 8 Apple G5 processors that are the same as used for the L1 trigger. These processors receive "trigger packets" or summaries independently from the Detached Vertex Trigger, the Stand-alone Muon trigger, and so-called Trigger primitives. Since the two main triggers can arrive at different times and even in different orders (all trigger primitives arrive at a fixed time relative to the beam crossing and have a maximum, very short latency), the system must buffer the trigger packets until it all packets from a crossing and then process them to make the final decision. Global Level 1 also can pre-scale various lower priority triggers and can do so "dynamically", that is based on the luminosity or the degree of congestion in the system. When GL1 decides that a beam crossing should be kept for subsequent processing by the Level 2/3 trigger, it issues a "Level 1 ACCEPT' to the Level 1 Buffer Memory that causes the crossing to be saved in a Level 2/3 buffer. Otherwise, the crossing will be deleted from the Level 1 Buffers (it is not clear yet whether BTeV will do this based on issuance of a Level 1 REJECT signal or just let the data be overwritten eventually in a circular buffer arrangement).

GL1 also maintains a list of the crossing numbers that have been accepted and receives requests for work from Level 2/3 processors. It responds to those requests by supplying a requesting CPU with a crossing number that

needs to be processed, thus initiating the Level 2/3 trigger calculation activities on that crossing by that processor.

GL1 maintains multiple trigger lists and provides the ability to partition the trigger and, in fact, the whole DA into subsections for commissioning and debugging purposes.

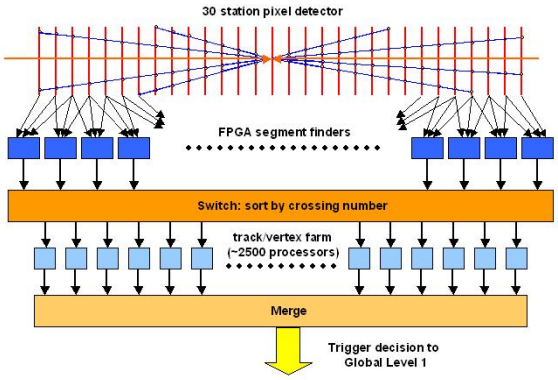

Figure 7: Schematic of the Level 1 hardware, showing the FPGA segment tracker, the sorting switch, the track/vertex processor and the merging of information into the Global Level 1 Trigger.

E. Highways

The original design of the BTeV system was felt to be difficult to implement because it required development of a high-speed *custom* switch. This was due to the need to handle data every 132 ns (the original crossing interval) and not due to the total throughput requirement. It is easy to underestimate the complexity and risk of a home-grown switch and associated software. *BTeV responded by dividing the system into 8 independent subsystems, called "highways", each handling only every 8th crossing. Since the system must deal with intervals of only 8x396 (132) ns, switches based on commercial networking gear will work!!!* However, all Data paths must be available to each highway, but now they can be lower speed links. There are eight times more slow links but their cost is about the same as for a single high speed link. We first implemented highways for event building into Level 2/3 and went to commercial network gear there. We have now demonstrated that we can use a commercial switch within Level 1 to sort the "track segments" according to crossing number. We now have the system divided into highways through all the trigger levels. Large amounts of home-grown hardware and software have thus been ELIMINATED resulting in reduced complexity and lower technical risk.

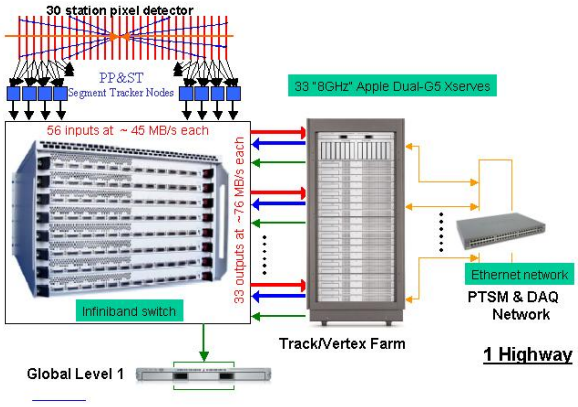

Figure 8: A possible realization of one highway of the Level 1 Trigger showing the use of an Infiniband Switch and commercial rack-mounted CPUs. There are 66 CPUs in the highway.

F. Level 1 Trigger Performance

The performance of the Level 1 Trigger is shown in Table 3.

Table 3: Efficiency of the Level 1 Trigger for (first entry) crossings containing only minimum bias events and (remaining entries) crossings also containing B decays

Process	Efficiency
Minimum Bias	1%
$B_s \rightarrow D^+_s K^-$	80%
$B^o \rightarrow J/\psi K_s$	65%
$B^o \rightarrow \phi K_s$	74%
B^o-> 2 body modes ($\pi^+\pi^-, K^+\pi^-, K^+K-$)	80%

The efficiency is calculated using simulated beam crossings with an average of six minimum bias events per crossing and one B event with the B decaying into the state indicated. For the efficiency on minimum bias events there is no B in the event. The event generator is PYTHIA [11] and GEANT3 [12] is used to perform a very detailed detector simulation. The calculated efficiency is the ratio of triggered events to all events that would have passed a "reasonable set" of "analysis cuts" to produce a clean signal with a good signal to background. Thus, it is the fraction of potentially useful events that is retained by the trigger.

The low efficiency on minimum bias events demonstrates that the trigger has good "rejection" for these unwanted events that constitute the vast majority of all interactions. The high efficiency on "potentially analysable" events containing B decays demonstrates the effectiveness of the Level 1 trigger to accept these events. This high efficiency is typical for a wide range of B decays spanning the whole range of B physics, including B baryon physics.

VII. LEVEL 2/3 TRIGGER

Level 2 (L2) and Level 3 (L3) triggers are implemented on the same PC Farm. When a processor from the farm becomes available it requests a crossing to analyse from the GL1. When it gets a crossing number, it broadcasts it to the buffer system and accumulates all the various fragments of data stored in the buffers. At present, all data is shipped even though only part of the detector information is used at Level 2. The network bandwidth is ample and "partial event readout" at Level 2 is considered an unnecessary complication. Once data is read into an Level 2/3 farm node, it is eligible for deletion from the data buffers and the only copy of it exists on the L2/3 processor. Figure 9 is a schematic representation of the L2/L3 trigger system.

In actuality, the input operations are overlapped with calculation so that a queue of several crossings is available to the processor at any time. This avoids the "deadtime" that would occur if a processor finished a calculation and had to wait until fresh data arrived.

L2 uses mainly tracking information from the pixel detector but also may use information from the forward silicon tracker and the forward straw tracker. It redoes the Level 1 calculation using the extra information and more refined algorithms to eliminate some false tracks that result in false triggers. This program must execute very quickly so it begins by reusing as much information as possible from the Level 1 calculations. The results of the Level 1 calculation are added to event buffer with the appropriate crossing number so that they are accessible to Level 2. A first implementation of the Level 2 program exists and achieves another factor of 10 reduction in uninteresting crossings while retaining about 90% of the interesting crossings, the ones containing B events. Its execution time is well below its time budget so it effectively satisfies the Level 2 requirements even at this first stage of development.

If a crossing satisfies the L2 trigger algorithm it is passed directly to the L3 process on the same node. No network operations or large memory-to-memory copies within the processor are required. If the crossing does not satisfy the Level 2 Trigger, the L2/3 process drops it, so it is lost forever. The processor then moves onto the next crossing.

The requirements on Level 3 are:

1. to obtain an additional factor of two rejection of unwanted crossings without hardly any loss (<5%) of good crossings;
2. to reduce the size of the data in the output of crossings accepted at Level 3 to approximately 80,000 bytes per accepted crossing. With an expected total of 2500 accepted events per second, this will restrict the average volume of output to 200 Mbytes/s. This will be done by partially reconstructing events and summarizing them, dropping some of the raw hits; and
3. to do as much reconstruction of the accepted crossings as possible to fully monitor every aspect of the detector and all subdetectors, to classify events according to the kind of B candidates that they might contain, and to reconstruct as much of the event as possible and to store the results of the calculations to get a head start on offline analysis.

Ideally, we would like Level 3 to do full event reconstruction and write DSTs (Data Summary "Tapes") from which physics analysis can be rapidly done. Since however we do not have a full Level 3 trigger algorithm or offline event reconstruction (although nearly every piece exists in a prototype form), we cannot be sure that we can in fact reconstruct every crossing completely. However, the base requirements listed above should be easily achievable based on what we already know. We do, for example, have a "tracking core" that is the main piece of code required to achieve the first two goals and it uses only 1/3 of the available CPU time.

The current size of the Level 2/3 farm, based on these considerations, is set at 1536 "12 GHz" equivalent processing units.

If the Level 3 trigger code decides to accept a crossing, it writes it out to archival storage by sending it through the same Level 2/3 network that it used to get the crossing data in the first place. However, the data is routed through a switch to the "data acquisition system" or DAQ. The DAQ will write the data to archival storage. If Level 3 rejects the crossing, it simply asks for new crossing to work on and the rejected crossing is eventually deleted from memory or overwritten and lost forever.

VIII. ADDITIONAL TRIGGER AND DAQ ISSUES

The DAQ will write the data to disk. In traditional systems, the data would eventually be copied to magnetic tape, usually to a mass storage system with automated, robotic, tape mounting capability. These systems usually present obstacles to efficient data access.

BTeV would prefer to avoid using magnetic tape and is exploring a "disk only" system with multiple copies distributed among physically separate sites for redundancy, backup, fail-over, and locality of access. These sites would all have access to high speed and high throughput networks so that they could work together to provide secure, robust, and responsive data access. Such a system presents great challenges but promises much easier and quicker access to the data than tape based systems.

Another important consideration is that the BTeV Level 2/3 Trigger Farm and DAQ have substantial internal disk buffers. There will be 300 Tbytes of disk storage within the Level 2/3 farm and the DAQ will have a "backing-store" of 1 petabyte of disk storage, enough to hold several months of data. This would permit buffering of crossings after Level 2 processing in the earliest, high luminosity part of the store, if the Level 3 could not keep up. As the luminosity declines during the store, the farm nodes could recall the unprocessed data from the disk store and complete the calculations.

Alternatively, data could be moved from the backing store to other resources, such as offline farms at Fermilab or at collaborating institutions to complete the processing.

The availability of inexpensive disk, large amounts of CPU power and excellent networking opens up many options for working in new ways. The Level 2/3 farm in fact provides BTeV with a very flexible computing platform to try out some of these novel approaches!

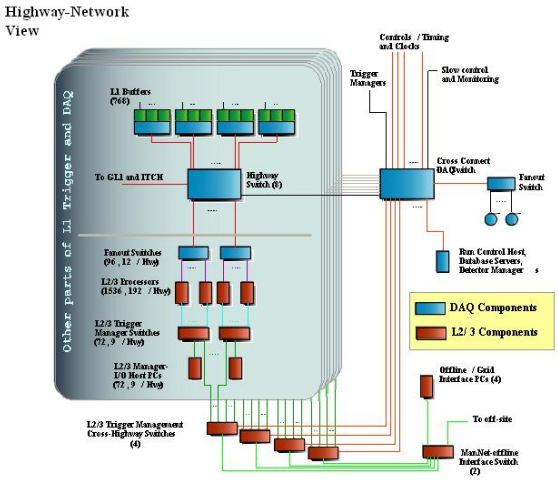

Figure 9: Schematic of the Level 2/3 Trigger system. The layers represent the 8 independent highways.

Another area of interest relates to the supervision and monitoring of the various Trigger System components. The system has supervisor nodes for each part of the trigger including a “Pixel Trigger Supervisor/Monitor” subsystem, a “Muon Trigger Supervisor//Monitor”, a “Global Trigger Supervisor/Monitor”, and a “Level 2/3 Supervisor and Monitor.” These talk to the DAQ and to the Run Control Subsystem. It is through these nodes that the systems are initialised, their configuration is modified during running, and the health of the system is monitored at a high level. A system of this complexity, however, needs an extraordinary level of fault handling and mitigation. This is discussed below in section X.

IX. Power and Cooling Issues

A new problem in the implementation of the BTeV trigger is the removal of the large amount of heat generated by the thousands of high power computing elements. Figure 10 shows the growth in the power per cabinet. These problems are now being confronted in computing centres and modern triggers have much in common with them. When the power density exceeds 7 KW per cabinet of 4U type enclosures, even traditional computer centre forced-air cooling systems with raised floor plenums cannot handle the power density [13]. This leads to hot spots and overheating in some areas of the cabinets. New solutions are being developed by industry. BTeV is waiting to see how this problem is solved and will adopt one of the successful approaches.

X. Fault Tolerance, Monitoring , and Mitigation

The BTeV Trigger System consists of hundreds of FPGAs, thousands of microprocessors, thousands of optical links, a highly complex network, terabytes of memory, and hundreds of terabytes of disk. These are almost all commercial parts. Given the expected mean time between failures in a system of this size, some components of the system will always be inoperative. Since system down time during collider stores inevitably results in data loss, the system must be designed to be highly fault tolerant. The system must be able to run with failures of single elements. Careful logging of problems will be necessary so that the impact of the failures on the physics results can be understood. Monitoring of system behavior to anticipate failures and schedule preventive measures is necessary. Thus, the system itself has to be to some degree self-aware. If significant failures occur, it must be possible to continue to operate with reduced functionality. It is highly desirable, probably even necessary, for the system to be “fault adaptive”, that is to be able to automatically adapt to problems with only limited operator intervention. Since conditions is the accelerator and the detector can change on very short times scales of minutes to hours the system must also be dynamically reconfigurable to achieve maximum performance.

BTeV has been involved in a research project, The Real Time Embedded System Project, or “RTES”[14], funded by the NSF to address the issue of fault tolerance and fault adaptation in large computing systems. The project involves BTeV physicists and computer scientists at Vanderbilt University, Syracuse University, University of Illinois, University of Pittsburgh, and Fermilab. It began in 2001 and will run for 5 years. While using the BTeV application to provide a concrete problem, the project addresses the general problem of reliability in large-scale clusters with real time constraints.

The basic approach of RTES is to develop and deploy a distributed, hierarchical fault detection and management system throughout the trigger and DAQ. The system is composed of hardware and software that function at various levels or layers of the system. At the lowest level, Very Light Agents (VLAs) monitor the performance of individual microprocessors and FPGAs for hardware failures and for software problems. Devices are organized into “farmlets” with a dedicated fault monitoring CPU for each of a group of trigger computing elements. The VLAs either handle problems locally and report mitigations to higher levels of the system for logging and trending purposes or, if they cannot mitigate the problem, report it to higher levels for assistance. Farmlets are grouped into “regions” and report to software called Adaptive Reconfigurable Mobile Objects of Reliability (ARMORs) running on regional fault management nodes. The regional nodes report to a Global Fault Manager. The system contains at the highest level a Generic Modelling Environment. This is used both to model the system behavior and generate software systems during

design and debugging and to provide a modelling and analysis framework for understanding the system during operations. The components of the system are shown in Fig. 10.

BTeV views this system as essential for its success. Significant software effort is devoted to developing it. Significant hardware resources, perhaps 10-15% of the total BTeV system, will be committed to this activity during operations.

XI. CONCLUSION

The BTeV Trigger System's unique feature is the use of massive computing based on "Commercial Off-the Shelf," or COTS, components at the lowest level and all subsequent levels. As such, it looks more like a normal computer farm than a conventional high energy physics trigger system. The rapid decrease in the cost of CPUs, memory, network equipment, and disk makes this approach feasible. The "highway" approach helps make this possible by reducing the maximum input rate seen by any one segment of the system. Custom chips are still required for the front end. Since on-the-fly sparsification is implemented there is no need for on-detector (short) pipelines and short, low latency Level 1 triggers that tend to constrain experiments in unpleasant ways. With the large number of components, the system must be made robust and fault tolerant. Significant resources must be provided to monitor the health of the system , record its status, and provide fault mitigation and remediation. In BTeV we expect to commit 10-15% of our hardware and maybe more of our effort on this problem. Infrastructure, especially power and cooling, is presenting new challenges. Because all systems have buffers that can receive simulated data, these systems can be completely debugged without beam. While successful implementation of this system is a big challenge, it is also an exciting one and the reward will be excellent science!

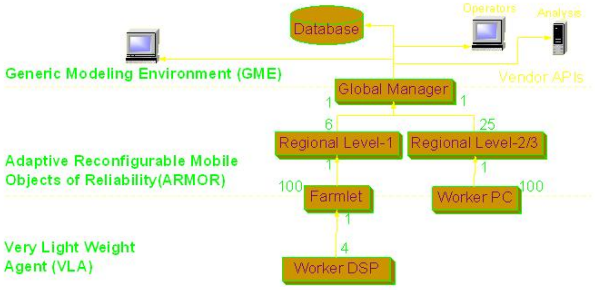

Figure 10: Schematic of the RTES system

XII. ACKNOWLEDGEMENTS

I would like to acknowledge the assistance of Erik Gottschalk and Michael Wang of Fermilab and of the BTeV trigger group in preparing this report. This work was supported in part by Fermilab, which is operated by Universities Research Association Inc. under Contract No. DE-AC02-76CH03000 with the United States Department of Energy.

XIII. REFERENCES

1) http://wwwbtev.fnal.gov/public/hep/general/proposal/index.shtml, ``BTeV Proposal Update"

2) E. E. Gottschalk, "Detached Vertex Trigger", Nucl. Inst. Meth. A473(2001)167

3) M.H.L.S. Wang representing the BTeV collaboration, "The BTeV Trigger System" in B Physics at Hadron Machines, 9th International Conference on B Physics at Hadron Machines, Beauty 2003. Pittsburgh, PA. Oct. 2003. AIP Conference Proc. Volume 722.

4) BTeV Technical Design Report, Chapter 11, The BTeV Trigger (contact erik@fnal.gov).

5) BTeV Technical Design Report, Chapter 12, Event Readout and Control System (contact votava@fnal.gov).

6) BTeV Technical Design Report, Chapter 4, The Pixel Vertex Detector (contact swalk@fnal.gov)

7) http://www.altera.com/products/devices/dev-index.jsp

8) http://www.xilinx.com

9) See, for example http://images.apple.com/server/pdfs/L301323A_XserveG5_TO.pdf

10) See, for example, http://searchstorage.techtarget.com/sDefinition/0,,sid5_gci214596,00.html; http://www.mellanox.com/products/hpcperformance.com; and http://lqcd.fnal.gov/benchmarks/newib/index.html

11) http://www.thep.lu.se/~torbjorn/Pythia.html

12) GEANT: CERN Program Library Long Writeup W5013, http://wwwinfo.cern.ch/asdoc/geant_html3/geantall.html

13) See http://www.datacenterdynamics.com/Portals/e5b81c2f-f780-4e59-88fc068dccab9568/SF04%20Sanmina.pdf

14) http://www-btev.fnal.gov/public/hep/detector/rtes/